Adversarial T-shirt! Evading Person Detectors in A Physical World

Kaidi Xu[1] Gaoyuan Zhang[2] Sijia Liu[2] Quanfu Fan[2] Mengshu Sun[1]
Hongge Chen[3] Pin-Yu Chen[2] Yanzhi Wang[1] Xue Lin[1]

[1]Northeastern University, USA
[2]MIT-IBM Watson AI Lab, IBM Research, USA
[3]Massachusetts Institute of Technology, USA

Abstract

It is known that deep neural networks (DNNs) are vulnerable to adversarial attacks. The so-called physical adversarial examples deceive DNN-based decision makers by attaching adversarial patches to real objects. However, most of the existing works on physical adversarial attacks focus on static objects such as glass frames, stop signs and images attached to cardboard. In this work, we propose Adversarial T-shirts, a robust physical adversarial example for evading person detectors even if it could undergo non-rigid deformation due to a moving person's pose changes. To the best of our knowledge, this is the first work that models the effect of deformation for designing physical adversarial examples with respect to non-rigid objects such as T-shirts. We show that the proposed method achieves 74% and 57% attack success rates in digital and physical worlds respectively against YOLOv2. In contrast, the state-of-the-art physical attack method to fool a person detector only achieves 18% attack success rate. Furthermore, by leveraging min-max optimization, we extend our method to the ensemble attack setting against two object detectors YOLO-v2 and Faster R-CNN simultaneously.

1. Introduction

The vulnerability of deep neural networks (DNNs) against adversarial attacks (namely, perturbed inputs deceiving DNNs) has been found in applications spanning from image classification to speech recognition [18, 34, 37, 6, 32, 1, 33]. Early works studied adversarial examples only in the digital space. Recently, some works showed that it is possible to create adversarial perturbations on physical objects and fool DNN-based decision makers under a variety of real-world conditions [28, 14, 2, 13, 25, 7, 30, 5, 21]. The design of *physical adversarial attacks* helps to evaluate the robustness of DNNs deployed in real-life systems, e.g., autonomous vehicles and surveillance systems. However, most of the studied physical adversarial attacks encounter two limitations: a) the physical objects are usually considered being *static*, and b) the possible *deformation* of adversarial pattern attached to a moving object (e.g., due to pose change of a moving person) is commonly neglected. In this paper, we propose a new type of physical adversarial attack, *adversarial T-shirt*, to evade DNN-based person detectors when a person wears the adversarial T-shirt; see the second row of Figure 1 for illustrative examples.

Related work Most of the existing physical adversarial attacks are generated against image classifiers and object detectors. In [28], a face recognition system is fooled by a real eyeglass frame designed under a crafted adversarial pattern. In [14], a stop sign is misclassified by adding black or white stickers on it against the image classification system. In [21], an image classifier is fooled by placing a crafted sticker at the lens of a camera. In [2], a so-called Expectation over Transformation (EoT) framework was proposed to synthesize adversarial examples robust to a set of physical transformations such as rotation, translation, contrast, brightness, and random noise. Compared to attacking image classifiers, generating physical adversarial attacks against object detectors is more involved. For example, the adversary is required to mislead the bounding box detector of an object when attacking YOLOv2 [26] and SSD [24]. A well-known success of such attacks in the physical world is the generation of adversarial stop sign [13], which deceives state-of-the-art object detectors such as YOLOv2 and Faster R-CNN [27].

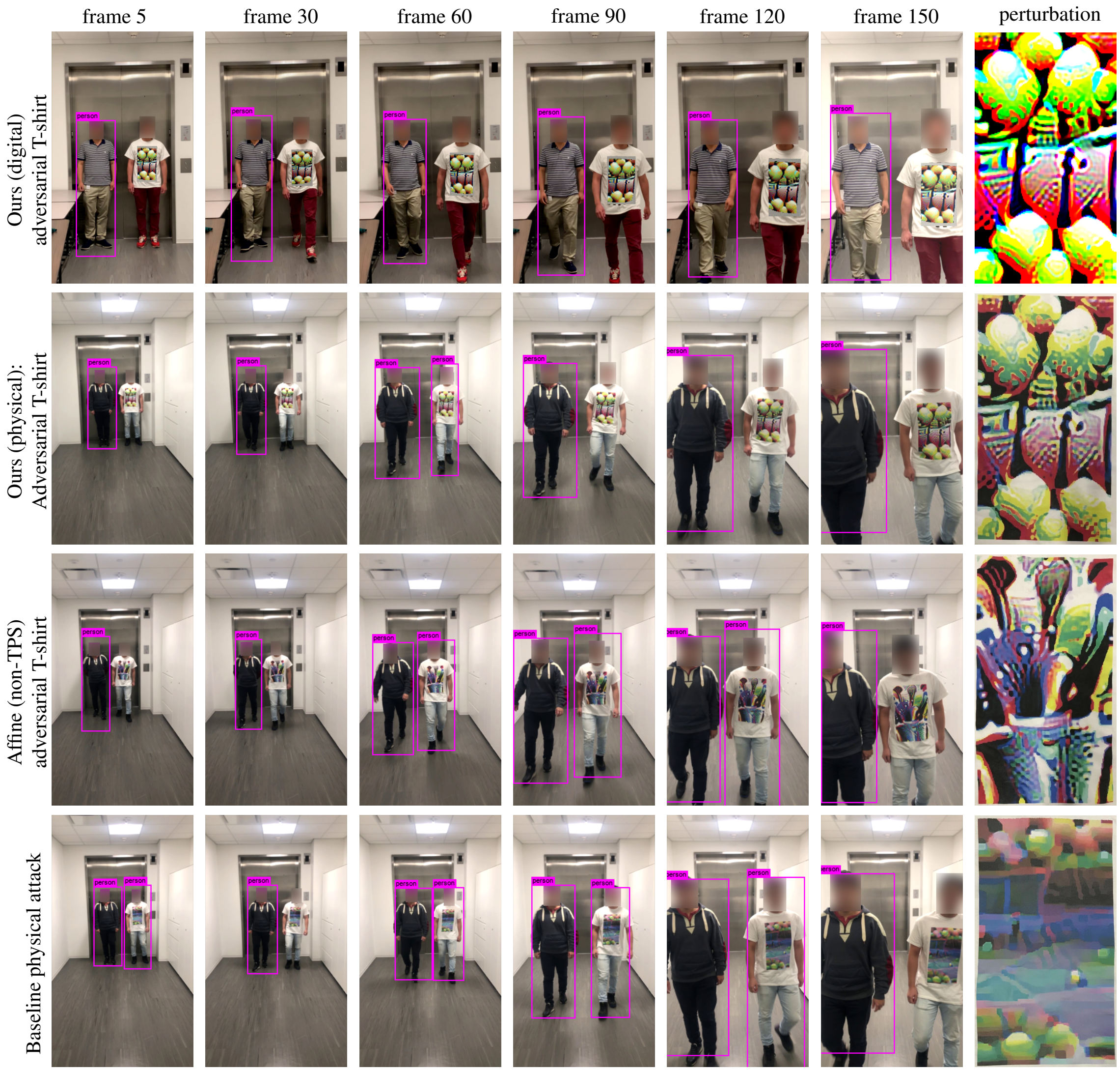


Figure 1: Evaluation of the effectiveness of adversarial T-shirts to evade person detection by YOLOv2. Each row corresponds to a specific attack method while each column except the last one shows an individual frame in a video. The last column shows the adversarial patterns applied to the T-shirts. At each frame, there are two persons, one of whom wears the adversarial T-shirt. First row: digital adversarial T-shirt generated using TPS. Second row: physical adversarial T-shirt generated using TPS. Third row: physical adversarial T-shirt generated using affine transformation (namely, in the absence of TPS). Fourth row: T-shirt with physical adversarial patch considered in [30] to evade person detectors.

The most relevant approach to ours is the work of [30], which demonstrates that a person can evade a detector by holding a cardboard with an adversarial patch. However, such a physical attack restricts the adversarial patch to be attached to a *rigid* carrier (namely, cardboard), and is different from our setting here where the generated adversarial pattern is directly printed on a T-shirt. We show that the attack proposed by [30] becomes ineffective when the adversarial patch is attached to a T-shirt (rather than a cardboard) and worn by a moving person (see the fourth row of Figure 1). At the technical side, different from [30] we propose a thin plate spline (TPS) based transformer to model deformation of non-rigid objects, and develop an ensemble physical attack that fools object detectors YOLOv2 and Faster R-CNN simultaneously. We highlight that our proposed adversarial T-shirt is not just a T-shirt with printed adversarial patch for clothing fashion, it is a physical adversarial wearable designed for evading person detectors in the real world.

Our work is also motivated by the importance of person detection on intelligent surveillance. DNN-based surveillance systems have significantly advanced the field of object detection [17, 16]. Efficient object detectors such as faster R-CNN [27], SSD [24], and YOLOv2 [26] have been deployed for human detection. Thus, one may wonder whether or not there

exists a security risk for intelligent surveillance systems caused by adversarial human wearables, e.g., adversarial T-shirts. However, paralyzing a person detector in the physical world requires substantially more challenges such as low resolution, pose changes and occlusions.

Contributions We summarize our contributions as follows:

- We develop a TPS-based transformer to model the temporal deformation of an adversarial T-shirt caused by pose changes of a moving person. We also show the importance of such non-rigid transformation to ensuring the effectiveness of adversarial T-shirts in the physical world.
- We propose a general optimization framework for design of adversarial T-shirts in both single-detector and multiple-detector settings.
- We conduct experiments in both digital and physical worlds and show that the proposed adversarial T-shirt achieves 74% and 64% attack success rates respectively when attacking YOLOv2. By contrast, the physical adversarial patch [30] printed on a T-shirt only achieves 27% attack success rate. Some of our results are highlighted in Figure 1.

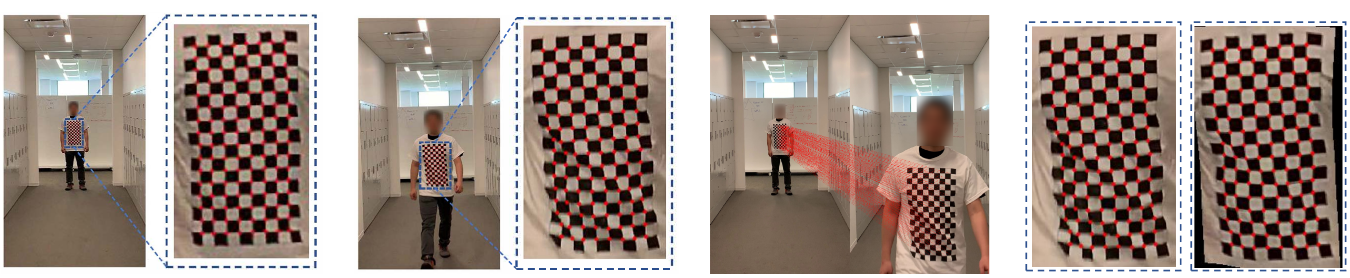
2. Modeling Deformation of A Moving Object by Thin Plate Spline Mapping

In this section, we begin by reviewing some existing transformations required in the design of physical adversarial examples. We then elaborate on the Thin Plate Spline (TPS) mapping we adopt in this work to model the possible deformation encountered by a moving and non-rigid object.

Let $\mathbf{x}$ be an original image (or a video frame), and $t(\cdot)$ be the physical transformer. The transformed image $\mathbf{z}$ under t is given by

$$\mathbf{z} = t(\mathbf{x}). \tag{1}$$

Existing transformations. In [2], the parametric transformers include scaling, translation, rotation, brightness and additive Gaussian noise; see details in [2, Appendix D]. In [23], the geometry and lighting transformations are studied via parametric models. Other transformations including perspective transformation, brightness adjustment, resampling (or image resizing), smoothing and saturation are considered in [29, 9]. All the existing transformations are included in our library of physical transformations. However, they are not sufficient to model the cloth deformation caused by pose change of a moving person. For example, the second and third rows of Figure 1 show that adversarial T-shirts designed against only existing physical transformations yield low attack success rates.



(a) (b) (c) (d)

Figure 2: Generation of TPS. (a) and (b): Two frames with checkerboard detection results. (c): Anchor point matching process between two frames (d): Real-world close deformation in (b) versus the synthesized TPS transformation (right plot).

TPS transformation for cloth deformation. A person's movement can result in significantly and constantly changing wrinkles (aka deformations) in her clothes. This makes it challenging to develop an adversarial T-shirt effectively in the real world. To circumvent this challenge, we employ TPS mapping [4] to model the cloth deformation caused by human body movement. TPS has been widely used as the non-rigid transformation model in image alignment and shape matching [19]. It

consists of an affine component and a non-affine warping component. We will show that the non-linear warping part in TPS can provide an effective means of modeling cloth deformation for learning adversarial patterns of non-rigid objects.

TPS learns a parametric deformation mapping from an original image $\mathbf{x}$ to a target image $\mathbf{z}$ through a set of control points with given positions. Let $\mathbf{p} := (\phi, \psi)$ denote the 2D location of an image pixel. The deformation from $\mathbf{x}$ to $\mathbf{z}$ is then characterized by the *displacement* of every pixel, namely, how a pixel at $\mathbf{p}^{(x)}$ on image $\mathbf{x}$ changes to the pixel on image $\mathbf{z}$ at $\mathbf{p}^{(z)}$, where $\phi^{(z)} = \phi^{(x)} + \Delta_\phi$ and $\psi^{(z)} = \psi^{(x)} + \Delta_\psi$, and Δ_ϕ and Δ_ψ denote the pixel displacement on image $\mathbf{x}$ along ϕ direction and ψ direction, respectively.

Given a set of n control points with locations $\{\hat{\mathbf{p}}_i^{(x)} := (\hat{\phi}_i^{(x)}, \hat{\psi}_i^{(x)})\}_{i=1}^n$ on image $\mathbf{x}$, TPS provides a parametric model of pixel displacement when mapping $\mathbf{p}^{(x)}$ to $\mathbf{p}^{(z)}$ [8]

$$\Delta(\mathbf{p}^{(x)}; \boldsymbol{\theta}) = a_0 + a_1 \phi^{(x)} + a_2 \psi^{(x)} + \sum_{i=1}^{n} c_i U(\|\hat{\mathbf{p}}_i^{(x)} - \mathbf{p}^{(x)}\|_2), \tag{2}$$

where $U(r) = r^2 \log(r)$ and $\boldsymbol{\theta} = [\mathbf{c}; \mathbf{a}]$ are the TPS parameters, and $\Delta(\mathbf{p}^{(x)}; \boldsymbol{\theta})$ represents the displacement along either ϕ or ψ direction.

Moreover, given the locations of control points on the transformed image $\mathbf{z}$ (namely, $\{\hat{\mathbf{p}}_i^{(z)}\}_{i=1}^n$), TPS resorts to a regression problem to determine the parameters $\boldsymbol{\theta}$ in (2). The regression objective is to minimize the distance between $\{\Delta_\phi(\mathbf{p}_i^{(x)}; \boldsymbol{\theta}_\phi)\}_{i=1}^n$ and $\{\hat{\Delta}_{\phi,i} := \hat{\phi}_i^{(z)} - \hat{\phi}_i^{(x)}\}_{i=1}^n$ along the ϕ direction, and the distance between $\{\Delta_\psi(\mathbf{p}_i^{(x)}; \boldsymbol{\theta}_\psi)\}_{i=1}^n$ and $\{\hat{\Delta}_{\psi,i} := \hat{\psi}_i^{(z)} - \hat{\psi}_i^{(x)}\}_{i=1}^n$ along the ψ direction, respectively. Thus, TPS (2) is applied to coordinate ϕ and ψ separately (corresponding to parameters $\boldsymbol{\theta}_\phi$ and $\boldsymbol{\theta}_\psi$). The regression problem can be solved by the following linear system of equations [10]

$$\begin{bmatrix} \mathbf{K} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0}_{3\times3} \end{bmatrix} \boldsymbol{\theta}_\phi = \begin{bmatrix} \hat{\boldsymbol{\Delta}}_\phi \\ \mathbf{0}_{3\times1} \end{bmatrix}, \begin{bmatrix} \mathbf{K} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0}_{3\times3} \end{bmatrix} \boldsymbol{\theta}_\psi = \begin{bmatrix} \hat{\boldsymbol{\Delta}}_\psi \\ \mathbf{0}_{3\times1} \end{bmatrix}, \tag{3}$$

where the (i,j)th element of $\mathbf{K} \in \mathbb{R}^{n\times n}$ is given by $K_{ij} = U(\|\hat{\mathbf{p}}_i^{(x)} - \hat{\mathbf{p}}_j^{(x)}\|_2)$, the ith row of $\mathbf{P} \in \mathbb{R}^{n\times 3}$ is given by $P_i = [1, \hat{\phi}_i^{(x)}, \hat{\psi}_i^{(x)}]$, and the ith elements of $\hat{\boldsymbol{\Delta}}_\phi \in \mathbb{R}^n$ and $\hat{\boldsymbol{\Delta}}_\psi \in \mathbb{R}^n$ are given by $\hat{\Delta}_{\phi,i}$ and $\hat{\Delta}_{\psi,i}$, respectively.

Non-trivial application of TPS The difficulty of implementing TPS for design of adversarial T-shirts exists from two aspects: 1) How to determine the set of control points? And 2) how to obtain positions $\{\hat{\mathbf{p}}_i^{(x)}\}$ and $\{\hat{\mathbf{p}}_i^{(z)}\}$ of control points aligned between a pair of video frames $\mathbf{x}$ and $\mathbf{z}$?

To address the first question, we print a *checkerboard* on a T-shirt and use the camera calibration algorithm [15, 36] to detect points at the intersection between every two checkerboard grid regions. These successfully detected points are considered as the control points of one frame. Figure 2-(a) shows the checkerboard-printed T-shirt, together with the detected intersection points. Since TPS requires a set of control points *aligned* between two frames, the second question on point matching arises. The challenge lies in the fact that the control points detected at one video frame are different from those at another video frame (e.g., due to missing detection). Figure 2-(a) v.s. (b) provides an example of point mismatch. To address this issue, we adopt a 2-stage procedure, *coordinate system alignment* followed by *point aliment*, where the former refers to conducting a perspective transformation from one frame to the other, and the latter finds the matched points at two frames through the nearest-neighbor method. We provide an illustrative example in Figure 2-(c). We refer readers to Appendix A for more details about our method.

3. Generation of Adversarial T-shirt: An Optimization Perspective

In this section, we begin by formalizing the problem of adversarial T-shirt and introducing notations used in our setup. We then propose to design a *universal* perturbation used in our adversarial T-shirt to deceive a *single* object detector. We lastly propose a min-max (robust) optimization framework to design the universal adversarial patch against *multiple* object detectors.

Let $\mathcal{D} := \{\mathbf{x}_i\}_{i=1}^M$ denote M video frames extracted from one or multiple given videos, where $\mathbf{x}_i \in \mathbb{R}^d$ denotes the ith frame. Let $\boldsymbol{\delta} \in \mathbb{R}^d$ denote the universal adversarial perturbation applied to $\mathcal{D}$. The adversarial T-shirt is then characterized by $M_{c,i} \circ \boldsymbol{\delta}$, where $M_{c,i} \in \{0,1\}^d$ is a bounding box encoding the position of the cloth region to be perturbed at the ith frame, and $\circ$ denotes element-wise product. *The goal of adversarial T-shirt is to design $\boldsymbol{\delta}$ such that the perturbed frames of $\mathcal{D}$ are mis-detected by object detectors.*

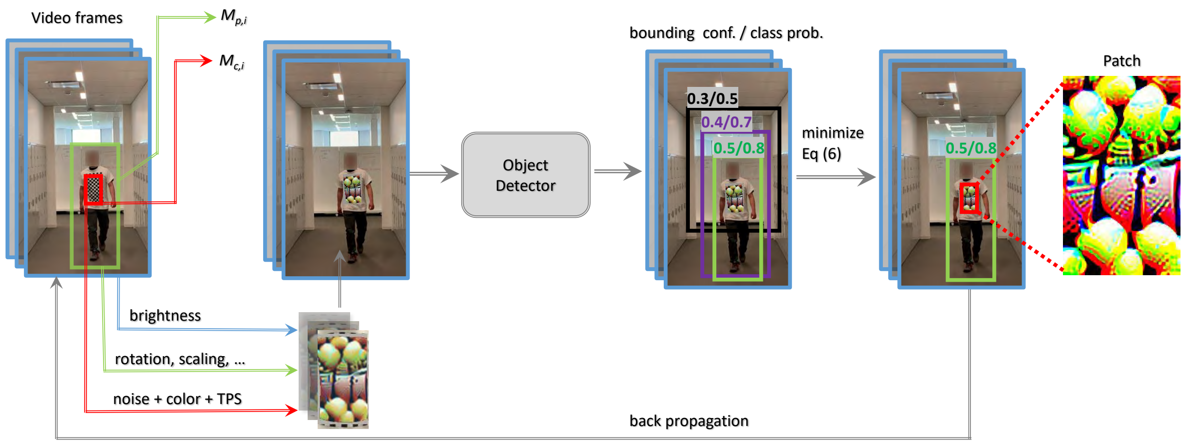


Figure 3: Overview of the pipeline to generate adversarial T-shirts. From left to right: First, the video frames containing a person whom wears the T-shirt with printed checkerboard pattern are used as training data. Second, the universal adversarial perturbation (to be designed) applies to the cloth region by taking into account different kinds of transformations. Third, the adversarial perturbation is optimized through problem (6) by minimizing the largest bounding-box probability belonging to the person class. The optimization procedure is performed as a closed loop through back-propagation.

Fooling a single object detector. We generalize the Expectation over Transformation (EoT) method in [3] for design of adversarial T-shirts. Note that different from the conventional EoT, a transformers' composition is required for generating an adversarial T-shirt. For example, a perspective transformation on the bounding box of the T-shirt is composited with an TPS transformation applied to the cloth region.

Let us begin by considering two video frames, an anchor image $\mathbf{x}_0$ (e.g., the first frame in the video) and a target image $\mathbf{x}_i$ for $i \in [M]$[1]. Given the bounding boxes of the person ($M_{p,0} \in \{0,1\}^d$) and the T-shirt ($M_{c,0} \in \{0,1\}^d$) at $\mathbf{x}_0$, we apply the perspective transformation from $\mathbf{x}_0$ to $\mathbf{x}_i$ to obtain the bounding boxes $M_{p,i}$ and $M_{c,i}$ at image $\mathbf{x}_i$. In the *absence* of physical transformations, the perturbed image $\mathbf{x}_i'$ with respect to (w.r.t.) $\mathbf{x}_i$ is given by

$$\mathbf{x}_i' = \underbrace{(\mathbf{1} - M_{p,i}) \circ \mathbf{x}_i}_{\text{A}} + \underbrace{M_{p,i} \circ \mathbf{x}_i}_{\text{B}} - \underbrace{M_{c,i} \circ \mathbf{x}_i}_{\text{C}} + \underbrace{M_{c,i} \circ \boldsymbol{\delta}}_{\text{D}}, \tag{4}$$

where the term A denotes the background region outside the bouding box of the person, the term B is the person-bounded region, the term C erases the pixel values within the bounding box of the T-shirt, and the term D is the newly introduced additive perturbation. In (4), the prior knowledge on $M_{p,i}$ and $M_{c,i}$ is acquired by person detector and manual annotation, respectively. Without taking into account physical transformations, Eq. (4) simply reduces to the conventional formulation of adversarial example $(1 - M_{c,i}) \circ \mathbf{x}_i + M_{c,i} \circ \boldsymbol{\delta}$.

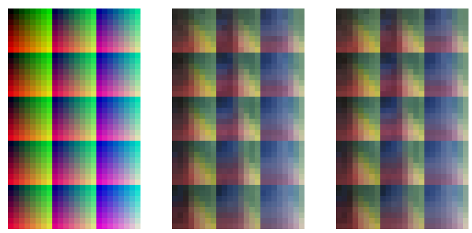
Next, we consider *three main types* of physical transformations: a) TPS transformation $t_{\text{TPS}} \in \mathcal{T}_{\text{TPS}}$ applying to the adversarial perturbation $\boldsymbol{\delta}$ for modeling the effect of cloth deformation, b) physical color transformation t_{color} which converts digital colors to those printed and visualized in the physical world, and c) conventional physical transformation $t \in \mathcal{T}$ applying to the region within the person's bounding box, namely, $(M_{p,i} \circ \mathbf{x}_i - M_{c,i} \circ \mathbf{x}_i + M_{c,i} \circ \boldsymbol{\delta})$. Here $\mathcal{T}_{\text{TPS}}$ denotes the set of possible non-rigid transformations, t_{color} is given by a regression model learnt from the color spectrum in the digital space to its printed counterpart, and $\mathcal{T}$ denotes the set of commonly-used physical transformations, e.g., scaling, translation, rotation, brightness, blurring and contrast. A modification of (4) under different sources of transformations is then given by

$$\mathbf{x}_i' = t_{\text{env}} \left(\text{A} + t\left(\text{B} - \text{C} + t_{\text{color}}(M_{c,i} \circ t_{\text{TPS}}(\boldsymbol{\delta} + \mu \mathbf{v}))\right)\right), \quad t \in \mathcal{T},\ t_{\text{TPS}} \in \mathcal{T}_{\text{TPS}}, \mathbf{v} \sim \mathcal{N}(0,1), \tag{5}$$

where the terms A, B and C have been defined in (4), and t_{env} denotes a brightness transformation to model the environmental brightness condition. In (5), $\mu \mathbf{v}$ is an additive Gaussian noise that allows the variation of pixel values, where μ is a given smoothing parameter and we set it as 0.03 in our experiments such that the noise realization falls into the range $[-0.1, 0.1]$. The randomized noise injection is also known as Gaussian smoothing [11], which makes the final objective function smoother and benefits the gradient computation during optimization.

[1] $[M]$ denotes the integer set $\{1, 2, \ldots, M\}$.

Remark 1 *The prior work, e.g., [28, 12], established a non-printability score (NPS) to measure the distance between the designed perturbation vector and a library of printable colors acquired from the physical world. The commonly-used approach is to incorporate NPS into the attack loss through regularization. However, it becomes non-trivial to find a proper regularization parameter, and the nonsmoothness of NPS makes optimization for the adversarial T-shirt difficult. To circumvent these challenges, we propose to model the color transformer* t_{color} *using a multilayer perception (MLP) of* 2 *hidden layers, each of which contains* 256 *and* 512 *neurons. Both the input and the output layers have dimension of* 3*. As shown in Figure 4, we generate the training dataset to map a digital color palette to the same one printed on a T-shirt. With the aid of* 960 *color cell pairs, we learn the weights of MLP by minimizing the mean squared error of the predicted physical color (with the digital color in Figure 4(a) as input) and the ground-truth physical color provided in Figure 4(b). Once the MLP-based color transformer* t_{color} *is learnt, we then incorporate it into (5).*



(a) (b) (c)

Figure 4: Physical color transformation. (a): The digital color map (b): The printed color map on a T-shirt (captured by the camera of iPhone X). (c): The predicted transformation from (a) via the learnt MLP.

With the aid of (5), the EoT formulation to fool a single object detector is cast as

$$\operatorname*{minimize}_{\boldsymbol{\delta}} \quad \frac{1}{M}\sum_{i=1}^{M} \mathbb{E}_{t,t_{\mathrm{TPS}},\mathbf{v}}\left[f(\mathbf{x}_i')\right] + \lambda g(\boldsymbol{\delta}) \tag{6}$$

where f denotes an attack loss for misdetection, g is the total-variation norm that enhances perturbations' smoothness [13], and $\lambda > 0$ is a regularization parameter. We further elaborate on our attack loss f in problem (6). In YOLOv2, a probability score associated with a bounding box indicates whether or not an object is present within this box. Thus, we specify the attack loss as the largest bounding-box probability over all bounding boxes belonging to the 'person' class. For Faster R-CNN, we attack all bounding boxes towards the class 'background'. The more detailed derivation on the attack loss is provided in Appendix B. Figure 3 presents an overview of our approach to generate adversarial T-shirts.

Min-max optimization for fooling multiple object detectors. Unlike digital space, the transferability of adversarial attacks largely drops in the physical environment, thus we consider a *physical ensemble attack* against multiple object detectors. It was recently shown in [31] that the ensemble attack can be designed from the perspective of min-max optimization, and yields much higher worst-case attack success rate than the averaging strategy over multiple models. Given N object detectors associated with attack loss functions $\{f_i\}_{i=1}^N$, the physical ensemble attack is cast as

$$\operatorname*{minimize}_{\boldsymbol{\delta}\in\mathcal{C}} \operatorname*{maximize}_{\mathbf{w}\in\mathcal{P}} \quad \sum_{i=1}^{N} w_i \phi_i(\boldsymbol{\delta}) - \frac{\gamma}{2}\|\mathbf{w} - \mathbf{1}/N\|_2^2 + \lambda g(\boldsymbol{\delta}), \tag{7}$$

where $\mathbf{w}$ are known as domain weights that adjust the importance of each object detector during the attack generation, $\mathcal{P}$ is a probabilistic simplex given by $\mathcal{P} = \{\mathbf{w} | \mathbf{1}^T\mathbf{w} = 1, \mathbf{w} \geq \mathbf{0}\}$, $\gamma > 0$ is a regularization parameter, and $\phi_i(\boldsymbol{\delta}) := \frac{1}{M}\sum_{i=1}^{M} \mathbb{E}_{t\in\mathcal{T},t_{\mathrm{TPS}}\in\mathcal{T}_{\mathrm{TPS}}}\left[f(\mathbf{x}_i')\right]$ following (6). In (7), if $\gamma = 0$, then the adversarial perturbation $\boldsymbol{\delta}$ is designed over the *maximum* attack loss (worst-case attack scenario) since $\mathrm{maximize}_{\mathbf{w}\in\mathcal{P}} \sum_{i=1}^{N} w_i\phi_i(\boldsymbol{\delta}) = \phi_{i^*}(\boldsymbol{\delta})$, where $i^* = \arg\max_i \phi_i(\boldsymbol{\delta})$ at a fixed $\boldsymbol{\delta}$. Moreover, if $\gamma \to \infty$, then the inner maximization of problem (7) implies $\mathbf{w} \to \mathbf{1}/N$, namely, an averaging scheme over M attack losses. Thus, the regularization parameter γ in (7) strikes a balance between the max-strategy and the average-strategy.

4. Experimental Results

In this section, we demonstrate the effectiveness of our approach for design of the adversarial T-shirt by comparing it with 2 attack methods, a) adversarial patch to fool YOLOv2 proposed in [30] and its printed version on a T-shirt (we call

baseline[2]), and b) the variant of our approach in the absence of TPS transformation, namely, $\mathcal{T}_{\mathrm{TPS}} = \emptyset$ in (5) (we call **affine**). We examine the convergence behavior of proposed algorithms as well as its Attack Success Rate[3] (ASR) in both digital and physical worlds. We clarify our algorithmic parameter setting in Appendix C.

Prior to detailed illustration, we briefly summarize the attack performance of our proposed adversarial T-shirt. When attacking YOLOv2, our method achieves 74% ASR in the digital world and 57% ASR in the physical world, where the latter is computed by averaging successfully attacked video frames over all different scenarios (i.e., indoor, outdoor and unforeseen scenarios) listed in Table 2. When attacking Faster R-CNN, our method achieves 61% and 47% ASR in the digital and the physical world, respectively. By contrast, baseline only achieves around 25% ASR in the best case among all digital and physical scenarios against either YOLOv2 or Faster R-CNN (e.g., 18% against YOLOv2 in the physical case). We also remark that compared to our earlier version [35], here the attack algorithm is updated by incorporating color transformations, and the physical test videos are taken under multiple scenes.

4.1. Experimental Setup

Data collection. We collect two datasets for learning and testing our proposed attack algorithm in digital and physical worlds. The training dataset contains 30 videos (1300 video frames), each of which takes 5-10 seconds and is captured by a moving person wearing a T-shirt with printed checkerboard under 4 different scenes: three indoor scenes and one outdoor scene. The desired adversarial pattern is then learnt from the training dataset. The test dataset in the digital space contains 10 videos captured under the same scenes as the training dataset. This dataset is used to evaluate the attack performance of the learnt adversarial pattern in the digital world. In the physical world, we customize a T-shirt with the printed adversarial pattern learnt from our algorithm. Another 24 test videos are then collected from two moving persons with one wearing the physical adversarial T-shirt. In addition, we also test our adversarial T-shirt by unforeseen scenarios, where the test videos involve different locations and different persons which are never covered in the training dataset. All videos are taken using an iPhone X and are resized to 416×416.

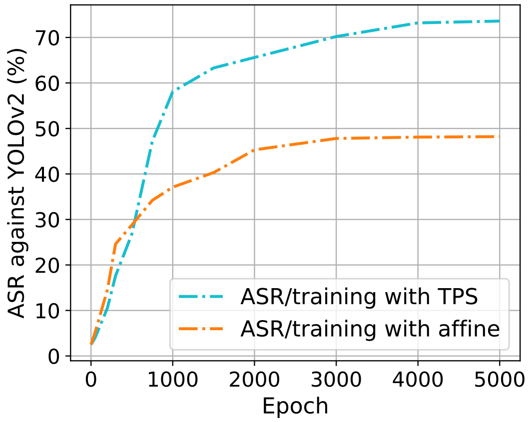
Object detectors. We use two state-of-the-art object detectors: Faster R-CNN [27] and YOLOv2 [26] to evaluate our method. These two object detectors are both pre-trained on COCO dataset [22] which contains 80 classes including 'person'. The detection minimum threshold are set as 0.7 for both Faster R-CNN and YOLOv2 by default.

4.2. Adversarial T-shirt in digital world

Convergence performance of proposed attack algorithm. In Figure 5, we show ASR against the epoch number used by our proposed algorithm to solve problem (6). Here the success of our attack at one testing frame is required to meet two conditions, a) misdetection of the person who wears the adversarial T-shirt, and b) successful detection of the person whom dresses a normal cloth. As we can see, the proposed attack method covnerges well for attacking both YOLOv2 and Faster R-CNN. We also note that attacking Faster R-CNN is more difficult than attacking YOLOv2. Furthermore, if TPS is not applied during training, then ASR drops around 30% compared to our approach by leveraging TPS.

ASR of adversarial T-shirts in various attack settings. We perform a more comprehensive evaluation on our methods in digital simulations. Table 1 compares ASR of adversarial T-shirts generated with or without TPS transformations in 4 attack settings: a) *single-detector attack* referring to adversarial T-shirts designed and evaluated using the same object detector, b) *transfer single-detector attack* referring to adversarial T-shirts designed and evaluated using different object detectors, c) *ensemble attack (average)* given by (7) but using the average of attack losses of individual models, and d) *ensemble attack (min-max)* given by (7). As we can see, it is crucial to incorporate TPS transformation in the design of adversarial T-shirts: ASR drops from 61% to 34% when attacking faster R-CNN and drops from 74% to 48% when attacking YOLOv2 in the single-detector attack setting. We also note that the transferability of single-detector attack is weak in all settings. And faster R-CNN is consistently more robust than YOLOv2, similar to the results in Figure 5. Compared to our approach and affine, the baseline method yields the worst ASR when attacking a single detector. Furthermore, we evaluate the effectiveness of the proposed min-max ensemble attack (7). As we can see, when attacking faster R-CNN, the min-max ensemble attack significantly outperforms its counterpart using the averaging strategy, leading to 15% improvement in ASR. This improvement is at the cost of 7% degradation when attacking YOLOv2.

[2]For fair comparison, we modify the perturbation size of baseline same as ours and execute the code provided in [30] under the same training dataset.
[3]ASR is given by the ratio of successfully attacked testing frames over the total number of testing frames.



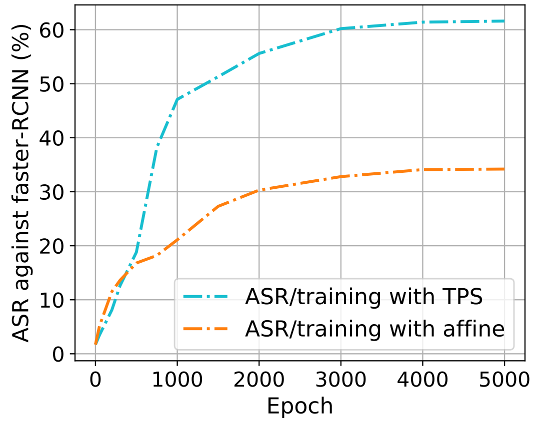


Figure 5: ASR v.s. epoch numbers against YOLOv2 (left) and Faster R-CNN (right).

Model \ Method	affine	ours (TPS)	baseline
(a) single-detector attack			
Faster R-CNN	34%	**61%**	22%
YOLOv2	48%	**74%**	24%
(b) transfer single-detector attack			
Faster-RCNN	9%	**10%**	9%
YOLOv2	12%	**13%**	10%
(c) ensemble attack (average)			
Faster-RCNN	16%	**32%**	NA
YOLOv2	31%	**60%**	NA
(d) ensemble attack (min-max)			
Faster-RCNN	32%	**47%**	NA
YOLOv2	27%	**53%**	NA

Table 1: The ASR (%) of adversarial T-shirts generated from our approach, affine and baseline under four attack settings in the digital world.

4.3. Adversarial T-shirt in physical world

We next evaluate our method in the physical world. First, we generate an adversarial pattern by solving problem (6) against YOLOv2 and Faster R-CNN, following Section 4.2. We then print the adversarial pattern on a white T-shirt, leading to the adversarial T-shirt. For fair comparison, we also print adversarial patterns generated by baseline and affine on white T-shirts of the same style.

At the testing phase, we use iPhone X to record videos for tracking a moving person wearing adversarial T-shirts in different scenarios. Different from taking static photos, our evaluation over an entire video takes into account multiple environment effects such as distance, deformation of the T-shirt, actions and angles of the moving person.

In Table 2, we compare our method with baseline [30] and affine under 3 specified scenarios, including the indoor scenario, outdoor scenario, and unforeseen scenario (indoor videos involve different locations and different persons which are never covered in the training dataset), together with the overall case of all scenarios. We observe that our method achieves 64% ASR (against YOLOv2), which is much higher than affine (39%) and baseline (19%) in the indoor scenario. Compared to the indoor scenario, evading person detectors in the outdoor scenario becomes more challenging. The ASR of our approach

reduces to 47% but outperforms affine (36%) and baseline (17%). This is not surprising since the outdoor scenario suffers more environmental variations. Even considering the unforeseen scenario, we find that our adversarial T-shirt is robust to the change of person and location, leading to 48% ASR against Faster R-CNN and 59% ASR against YOLOv2. Compared to the digital results, the ASR of our adversarial T-shirt drops around 10% in all tested physical-world scenarios.

Model \ Method	affine	ours (TPS)	baseline
indoor scenario			
Faster R-CNN	27%	**50%**	15%
YOLOv2	39%	**64%**	19%
outdoor scenario			
Faster R-CNN	25%	**42%**	16%
YOLOv2	36%	**47%**	17%
unforeseen scenario			
Faster R-CNN	25%	**48%**	12%
YOLOv2	34%	**59%**	17%
all scenarios			
Faster R-CNN	26%	**47%**	14%
YOLOv2	37%	**57%**	18%

Table 2: The ASR (%) of adversarial T-shirts generated from our approach, affine and baseline under different physical-world scenarios.

In Figure 6, we demonstrate our physical-world attack results in two scenarios: a) adversarial T-shirts generated by our method, affine and baseline in an outdoor scenario (the first three rows), b) adversarial T-shirts generated by our method and affine in an unforeseen scenario (at a location never seen in the training dataset). As we can see, our method outperforms affine and baseline. In the absence of TPS, adversarial T-shirts generated by affine and baseline fail in most of cases, implying the importance of TPS to model the T-shirt deformation. When a person whom wears the adversarial T-shirt walks towards the camera, as expected, the detector also becomes easier to be attacked.

5. Conclusion

In this paper, we propose *Adversarial T-shirt*, the first successful adversarial wearable to evade detection of moving persons. Since T-shirt is a non-rigid object, its deformation induced by a person's pose change is taken into account when generating adversarial perturbations. We also propose a min-max ensemble attack algorithm to fool multiple object detectors simultaneously. We show that our attack against YOLOv2 can achieve 74% and 57% attack success rate in the digital and physical world, respectively. By contrast, the baseline method can only achieve 24% and 18% ASR. Based on our studies, we hope to provide some implications on how the adversarial perturbations can be implemented with human clothing, accessories, paint on face, and other wearables, and we also aim to establish a general framework for evaluating the robustness of real-time machine learning systems deployed in physical worlds.

References

[1] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *arXiv preprint arXiv:1802.00420*, 2018.

[2] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok. Synthesizing robust adversarial examples. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 284–293, 10–15 Jul 2018.

[3] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *International Conference on Machine Learning*, pages 284–293, 2018.

[4] Fred L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on pattern analysis and machine intelligence*, 11(6):567–585, 1989.

[5] Yulong Cao, Chaowei Xiao, Dawei Yang, Jing Fang, Ruigang Yang, Mingyan Liu, and Bo Li. Adversarial objects against lidar-based autonomous driving systems. *arXiv preprint arXiv:1907.05418*, 2019.

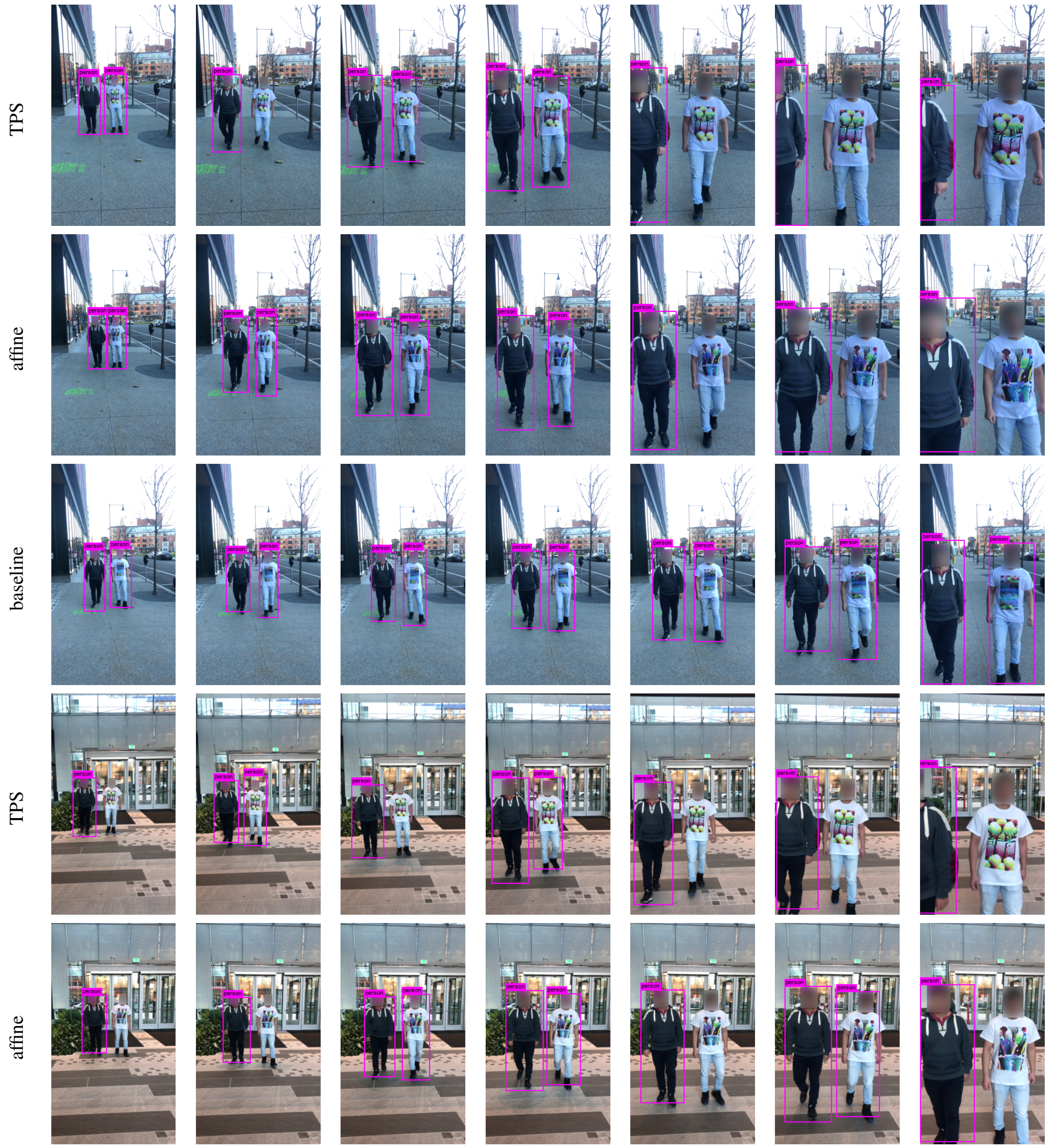


Figure 6: Some testing frames in the physical world using adversarial T-shirt against YOLOv2. All frames are performed by two persons with one wearing the proposed adversarial T-shirt, generated by our method (TPS), affine and baseline. The first three rows: an outdoor scenario. The last two rows: an unforeseen scenario.

[6] Nicholas Carlini and David Wagner. Audio adversarial examples: Targeted attacks on speech-to-text. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 1–7. IEEE, 2018.

[7] Shang-Tse Chen, Cory Cornelius, Jason Martin, and Duen Horng Polo Chau. Shapeshifter: Robust physical adversarial attack on faster r-cnn object detector. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 52–68. Springer, 2018.

[8] Haili Chui. Non-rigid point matching: algorithms, extensions and applications. *Citeseer*, 2001.

[9] Gavin Weiguang Ding, Kry Yik Chau Lui, Xiaomeng Jin, Luyu Wang, and Ruitong Huang. On the sensitivity of adversarial robustness to input data distributions. In *International Conference on Learning Representations*, 2019.

[10] Gianluca Donato and Serge Belongie. Approximate thin plate spline mappings. In *European conference on computer vision*, pages 21–31. Springer, 2002.

[11] John C Duchi, Peter L Bartlett, and Martin J Wainwright. Randomized smoothing for stochastic optimization. *SIAM Journal on Optimization*, 22(2):674–701, 2012.

[12] Ivan Evtimov, Kevin Eykholt, Earlence Fernandes, Tadayoshi Kohno, Bo Li, Atul Prakash, Amir Rahmati, and Dawn Song. Robust physical-world attacks on machine learning models. *arXiv preprint arXiv:1707.08945*, 2017.

[13] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Florian Tramer, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Physical adversarial examples for object detectors. In *12th USENIX Workshop on Offensive Technologies (WOOT 18)*, 2018.

[14] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song. Robust physical-world attacks on deep learning visual classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1625–1634, 2018.

[15] Andreas Geiger, Frank Moosmann, Ömer Car, and Bernhard Schuster. Automatic camera and range sensor calibration using a single shot. In *2012 IEEE International Conference on Robotics and Automation*, pages 3936–3943. IEEE, 2012.

[16] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[17] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[18] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[19] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, pages 2017–2025, 2015.

[20] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *2015 ICLR*, arXiv preprint arXiv:1412.6980, 2015.

[21] Juncheng Li, Frank Schmidt, and Zico Kolter. Adversarial camera stickers: A physical camera-based attack on deep learning systems. In *International Conference on Machine Learning*, pages 3896–3904, 2019.

[22] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[23] Hsueh-Ti Derek Liu, Michael Tao, Chun-Liang Li, Derek Nowrouzezahrai, and Alec Jacobson. Beyond pixel norm-balls: Parametric adversaries using an analytically differentiable renderer. In *International Conference on Learning Representations*, 2019.

[24] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[25] Jiajun Lu, Hussein Sibai, and Evan Fabry. Adversarial examples that fool detectors. *arXiv preprint arXiv:1712.02494*, 2017.

[26] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[27] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[28] Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael K Reiter. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 1528–1540. ACM, 2016.

[29] C. Sitawarin, A. N. Bhagoji, A. Mosenia, P. Mittal, and M. Chiang. Rogue signs: Deceiving traffic sign recognition with malicious ads and logos. *arXiv preprint arXiv:1801.02780*, 2018.

[30] Simen Thys, Wiebe Van Ranst, and Toon Goedemé. Fooling automated surveillance cameras: adversarial patches to attack person detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[31] Jingkang Wang, Tianyun Zhang, Sijia Liu, Pin-Yu Chen, Jiacen Xu, Makan Fardad, and Bo Li. Beyond adversarial training: Min-max optimization in adversarial attack and defense. *arXiv preprint arXiv:1906.03563*, 2019.

[32] Kaidi Xu, Hongge Chen, Sijia Liu, Pin-Yu Chen, Tsui-Wei Weng, Mingyi Hong, and Xue Lin. Topology attack and defense for graph neural networks: An optimization perspective. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2019.

[33] Kaidi Xu, Sijia Liu, Gaoyuan Zhang, Mengshu Sun, Pu Zhao, Quanfu Fan, Chuang Gan, and Xue Lin. Interpreting adversarial examples by activation promotion and suppression. *arXiv preprint arXiv:1904.02057*, 2019.

[34] Kaidi Xu, Sijia Liu, Pu Zhao, Pin-Yu Chen, Huan Zhang, Quanfu Fan, Deniz Erdogmus, Yanzhi Wang, and Xue Lin. Structured adversarial attack: Towards general implementation and better interpretability. In *International Conference on Learning Representations*, 2019.

[35] Kaidi Xu, Gaoyuan Zhang, Sijia Liu, Quanfu Fan, Mengshu Sun, Hongge Chen, Pin-Yu Chen, Yanzhi Wang, and Xue Lin. Evading real-time person detectors by adversarial t-shirt. *arXiv preprint arXiv:1910.11099*, 2019.

[36] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22, 2000.

[37] Pu Zhao, Kaidi Xu, Sijia Liu, Yanzhi Wang, and Xue Lin. Admm attack: an enhanced adversarial attack for deep neural networks with undetectable distortions. In *Proceedings of the 24th Asia and South Pacific Design Automation Conference*, pages 499–505. ACM, 2019.

Appendix

In the supplement, we provide details on the thin plate spline (TPS) transformation, the formulation of attack loss, the setting of algorithmic parameters, and the additional experiments of the adversarial T-shirt in the physical world.

A. How to construct TPS transformation?

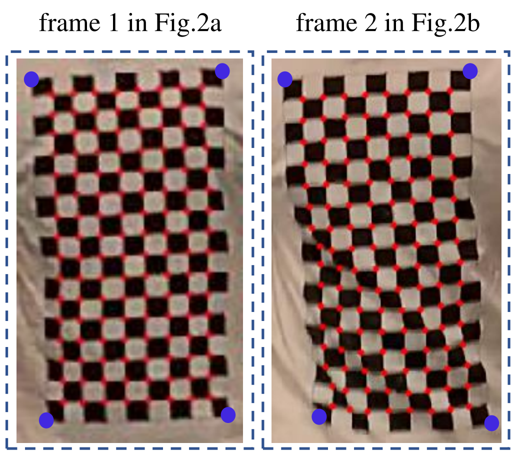


Figure A1: Four manually annotated corner points (blue) used to generate the bounding box of cloth region at frame i, namely, $M_{c,i}$. And 8×16 anchor points (red) on the checkerboard used to generate TPS transformation t_{TPS} between two video frames.

We first manually annotate four corner points (see blue markers in Figure A1) to conduct a perspective transformation between two frames at different time instants. This perspective transformation is used to align the coordinate system of anchor points used for TPS transformation between two frames.

Ideally, the checkerboard detection tool [15, 36] always outputs a grid of corner points detected. In most cases, it can locate all the 8×16 points on the checkerboard perfectly, so no additional effort is needed to establish the point correspondences between two images. In the case when there are corner points missing in the detection, we use the following method to match two images. We perform a point matching procedure (see Algorithm 1) to align the anchor points (see red markers in Figure A1) detected by the checkerboard detection tool. The data matching procedure selects the set of matched anchor points used for constructing TPS transformation.

Algorithm 1 Constructing TPS transformation

Input: Given original image $\mathbf{x}_1$ (frame 1) with $r_1 \times c_1$ anchor points, each of which has coordinate $\mathbf{p}^{(1)}[i,j]$, where $i \in [r_1]$, $j \in [c_1]$ and $[n]$ denotes the integer set $\{1, 2, \ldots, n\}$, target image $\mathbf{x}_2$ (frame 2) with $r_2 \times c_2$ anchor points, each of which has coordinate $\mathbf{p}^{(2)}[i,j]$, where $i \in [r_2]$ and $j \in [c_2]$, distance tolerance $\epsilon > 0$, and empty vectors $\tilde{\mathbf{p}}^{(1)}$ and $\tilde{\mathbf{p}}^{(2)}$.

Output: Matched $r \times c$ anchor points $\tilde{\mathbf{p}}^{(1)}[i,j]$ versus $\tilde{\mathbf{p}}^{(2)}[i,j]$ for $i \in [r]$ and $j \in [c]$, and TPS transformation t_{TPS} from $\mathbf{x}_1$ to $\mathbf{x}_2$.

for $(i,j) \in [r_1] \times [c_1]$ **do**
    given $\mathbf{p}^{(1)}[i,j]$ in $\mathbf{x}_1$, find the candidate of matching point $\mathbf{p}^{(2)}[i',j']$ by nearest neighbor in $\mathbf{x}_2$,
    if $\|\mathbf{p}^{(1)}[i,j] - \mathbf{p}^{(2)}[i',j']\|_2 \le \epsilon$ **then**
        matching $\mathbf{p}^{(1)}[i,j]$ with $\mathbf{p}^{(2)}[i',j']$, and adding them into $\tilde{\mathbf{p}}^{(1)}$ and $\tilde{\mathbf{p}}^{(2)}$ respectively,
    end if
end for
build TPS transformation t_{TPS} by solving Eq. (2) given $\tilde{\mathbf{p}}^{(1)}$ and $\tilde{\mathbf{p}}^{(2)}$.

B. Formulation of attack loss

There are two possible options to formulate the attack loss f to fool person detectors. First, f is specified as the misclassification loss, commonly-used in most of previous works. The goal is to misclassify the class 'person' to any other incorrect class. However, our work consider a more advanced disappearance attack, which enforces the detector even not to draw the bounding box of the object 'person'. For YOLOv2, we minimize the confidence score of all bounding boxes corresponding to the class 'person'. For Faster R-CNN, we attack all bounding boxes towards the class 'background'. Let $\mathbf{x}_i'$ be a perturbed video frame, the attack loss in (6) is then given by

$$f(\mathbf{x}_i') = \max_j \{\max\{p_j(\mathbf{x}_i'), \nu\} \cdot \mathbb{1}_{|B_j \cap M_{p,i}| > \eta}\}, \tag{8}$$

where $p_j(\mathbf{x}_i')$ denotes the confidence score of the jth bounding box for YOLOv2 or the probability of the 'person' class at the jth bounding box for Faster R-CNN, ν is a confidence threshold, the use of $\max\{p_j(\mathbf{x}_i'), \nu\}$ enforces the optimizer to minimize the bounding boxes of high probability (greater than ν), B_j is the jth bounding box, $M_{p,i}$ is the known bounding box encoding the person's region, the quantity $|B_j \cap M_{p,i}|$ represents the intersection between B_j and $M_{p,i}$, $|\cdot|$ is the cardinality function, and $\mathbb{1}_{|B_j \cap M_{p,i}| > \eta}$ is the indicator function, which returns 1 if B_j has at least η-overlapping with $M_{c,i}$, and 0 otherwise. In Eq.(8), the quantity $\max\{p_j(\mathbf{x}_i'), \nu\} \cdot \mathbb{1}_{|B_j \cap M_{p,i}| > \eta}$ characterizes the bounding box of our interest with both high probability and large overlapping with $M_{p,i}$. And the eventual loss in Eq.(8) gives the largest probability for detecting a bounding box of the object 'person'.

C. Hyperparameter setting

When solving Eq. (6), we use Adam optimizer [20] to train 5,000 epochs with the initial learning rate, 1×10^{-2}. The rate is decayed when the loss ceases to decrease. The regularization parameter λ for total-variation norm is set as 3. In Eq. (7), we set γ as 1, and solve the min-max problem by 6000 epochs with the initial learning rate 1×10^{-2}. In Eq. (5), the details of transformations t are shown in Table A1.

Transformation	Minimum	Maximum
Scale	0.5	2
Brightness	-0.1	0.1
Contrast	0.8	1.2
Random uniform noise	-0.1	0.1
Blurring	average pooling/filter size = 5	

Table A1: The conventional transformations t in Eq. (5).

In experiments, we find that the hyperparameter λ strikes a balance between the fine-gained perturbation pattern and its smoothness. As we can see in Figure A2, when λ is smallest (namely, $\lambda = 1$), the perturbation can achieve the best ASR (82%) against YOLOv2 in the digital space, however when we test the digital pattern in the physical world, the attacking performance drops to 51% (worse than the case of $\lambda = 3$) as the non-smooth (sharp) perturbation pattern might not be well captured by a real-world camera. In our experiments, we choose $\lambda = 3$ for the best tradeoff between digital and physical results.

For a real-world deployment of a person detector, the minimum detection threshold needs to be empirically determined to obtain a good tradeoff between detection accuracy and false alarm rates. In our physical-word testing, we set the threshold to 0.7 for Faster R-CNN and YOLOv2, at which both of them achieve detection accuracy over 97% on person wearing normal clothing. The sensitivity analysis of this threshold is provided in Figure A3.

λ	1	3	5
			
digital	82%	74%	69%
physical	51%	57%	55%

Figure A2: ASR v.s. λ against YOLOv2.

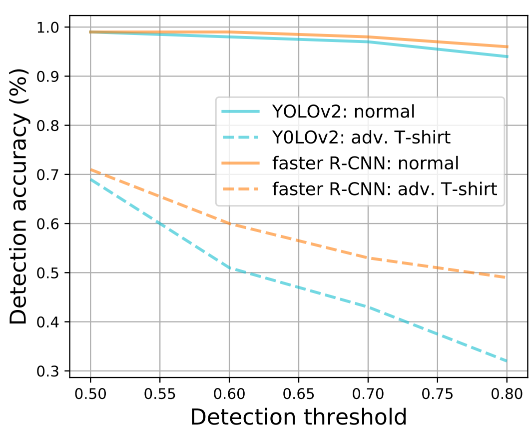


Figure A3: The detection accuracy of YOLOV2 and Faster R-CNN under different detection thresholds . ‘Normal’ means the case of persons wearing normal clothing, and ‘adv. T-shirt’ means the case of persons wearing the adversarial T-shirt.